The *e-Motion* Team-Project

« Geometry and Probability for motion and action »

Inria Rhône-Alpes & Gravir laboratory (UMR 5527)

Scientific leader : Christian LAUGIER

Team-Project members *(year 2004)*

- **Permanent staff**
  - Christian Laugier, DR2 Inria *(Scientific leader)*
  - Emmanuel Mazer, DR2 Cnrs *(External collaborator currently in our start-up « Probayes »)*
  - Pierre Bessiere, CR1 Cnrs
  - Thierry Fraichard, CR1 Inria
  - Sepanta Sekhavat, CR1 Inria *(Currently in Iran for 2 years)*
  - Olivier Aycard, MC UJF
  - Anne Spalanzani, MC UPMF

- **Invited researchers, Postdocs, and Engineers**
  - Juan-Manuel Ahuactzin
  - Olivier Malrait
  - Kamel Mekhnacha
  - Jorge Hermosillo
  - Christophe Coué

- **PhD students**
  - 7 theses defended in 2003: *J. Diard, C. Mendoza, R. Garcia, J. Hermosillo, F. Large, C. Coué, K. Sundaraj*
  - 8 PhD students: *C. Pradalier, C. Koike, M. Amavicza, R. Lehy, F. Colas, D. Vasquez, P.Dangauthier, M. Yguel*
  - 3 PhD students co-directed : *M. Kais (rocquencourt), S. Petti (rocquencourt), B. Rebsamen (NUS)*
  - 5 Master students : *Christopher Tay, Alejandro xx, Ruth xx, Christophe Braillon, Julien Burlet*

Objectives & motivations

• **Scientific challenge :** *To develop new models for constructing « artificial systems » having sensing, decisional, and acting capabilities sufficiently efficient and robust for making them really operational in open (i.e. large & weakly structured) and dynamic environments.*

• **Practical objective :** *To built operational (i.e. scalable) systems in some selected application domains (e.g. transportation, personal robots …). Such systems should be able to « share our living space while exhibiting natural behaviors ».*

• **Motivations & difficulties :** *Instead of promises & impressive advances in robotics in the last decade, almost no advanced robots are currently evolving around us!*
⇒ No reliability, weak reactivity, low efficiency (real time constraints), cybernetic behaviors, programming difficulty & no real learning capabilities.

• **Favourable technological context :** *(1) Continuous & fast growing of computational power; (2) Fast development of micro & nano technologies (mechatronics); (3) Increasing impact of information & telecom technologies on our everyday life (ambiant intelligence)*

Cooperation & Contracts

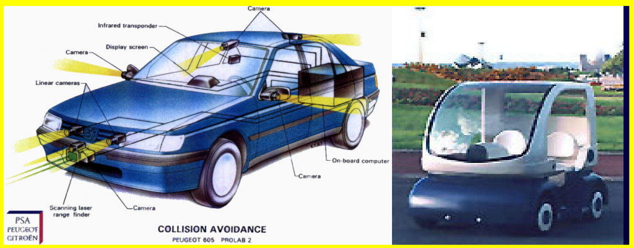
- **International cooperation**
  Japan (Riken), Singapore (NTU, NUS, SIMTech), USA (UCLA, Stanford, MIT), Mexico (IESTM Monterrey, UDLA Puebla), Europe (EPFL, Univ college of London)
- **Industrial cooperation**
  Robosoft, Renault, PSA, XL-Studio, Aesculap-BBrown, Teamlog, Kelkoo
  Startups: ITMI, Getris Images, Aleph Technologies, Aleph Med, Probayes (oct. 2003)
- **R&D contracts**
  - Industrial : *ACI « Protege », Priamm « Visteo », RNTL « Amibe », Kelkoo*
  - National : *Robea (EVbayes, Parknav), Predit (Arcos, MobiVip, Puvame)*
  - Europe : *NoE « Euron », IST-FET « Biba », IST « Cybercars », IST « Prevent»*

A large spectrum of potential applications

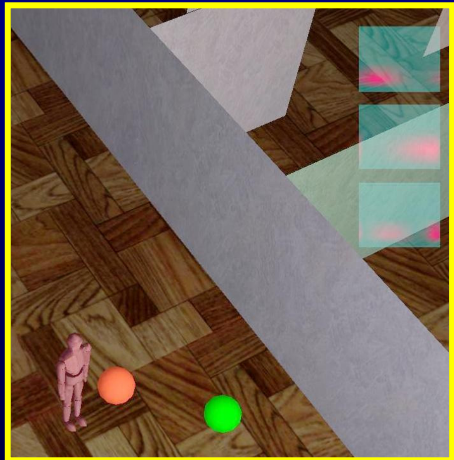
=> Rehabilitation & Medical care, Services, Transport & Logistics, Entertainment …

Main considered applications

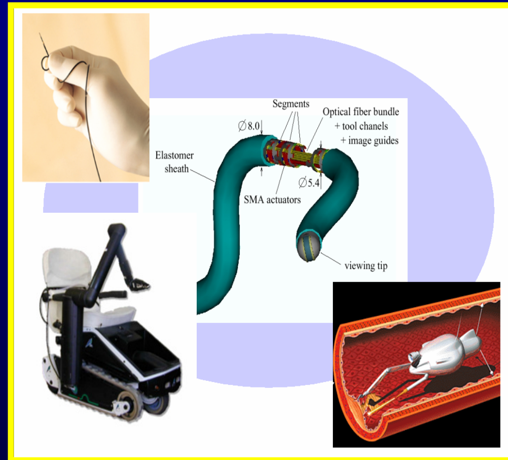
"Personal Robot Assistant"



Future cars
(driving assistance & autonomous driving)



Virtual autonomous agents
(natural interaction with humen)



Rehabilitation & Medical robots

Scientific approach

- **Main difficulties**
  - **Previous approaches on AI & Robotics have shown their limitations**
    - *=> Logics (70's), Geometry (80's), Random search (90's), purely Reactive Architectures (90's)*
  - **The real world is too complex for being fully modeled using classical tools (in particular: incompleteness & uncertainty)**
    - *=> Additional methods are required (e.g. probabilistic programming)*
    - *=> Biologic inspiration could bring some help (sensori-motors systems, internal representations for motions… which seems mostly based on probabilistic laws)*

- **Required technological breakthroughs**
  - **Motion & action autonomy in a complex dynamic world**
    - *=> Incremental world modeling, time space dimension, prediction & estimation of obstacles motion*
  - **Increased robustness & safety of navigation systems (perception & control)**
    - *=> Dealing with incompleteness & uncertainty*
  - **Easy programming & system adaptativity**
    - *=> Self learning capabilities & behaviors coding and mixing*

- **Our approach**
  - *⇒ To focus on « complexity » and « incompleteness » problems (scalability & real world)*
  - *⇒ We guess that this can be achieved by combining geometrical and probabilistic approache*

Two complementary reasoning processes

Dynamic world

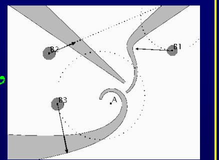
Analytical & Statistical data
Sensing data

Geometric & Kinematic reconstruction,
SLAM
Motion prediction

Space & Motion Models

Constrained Motion Planning

Differential Flatness,
Velocity Space, ITP



Motion plan & Navigation controls

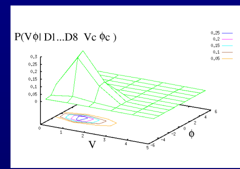
Mastering the complexity by using the right reasoning level & incremental approaches

Incompleteness

Preliminary Knowledge
+
Experimental Data
=
Probabilistic Representation

Maximum Entropy
Principle

$$-\sum P_i \log(P_i)$$



Uncertainty

Bayesian Inference
(NP-Hard [Cooper 90]
Heuristics & Optimization)

P(AB|C)=P(A|C)P(B|AC)=P(B|C)P(A|BC)
P(A|C)+P(¬A|C) = 1

Queries & Decision Process

Taking explicitly into account the hidden variables at the reasoning level

Bayesian Programming (principle)

Bayesian Program

Description

- Specification => *Preliminary Knowledge* π
  - *Set of relevant variables* $\{X^1, X^2, \ldots, X^n\}$
  - *Decomposition of the joint distribution (efficient way to compute it)*

$$\mathrm{P}\left(X^1 \otimes X^2 \otimes \ldots \otimes X^n \mid \delta \otimes \pi\right) = \mathrm{P}\left(L^1 \mid \delta \otimes \pi\right) \times \mathrm{P}\left(L^2 \mid R^2 \otimes \delta \otimes \pi\right) \times \ldots \times \mathrm{P}\left(L^k \mid R^k \otimes \delta \otimes \pi\right)$$

  - *Parametric forms assigned to some of the terms appearing in the decomposition (gaussian, uniform …)* $\mathrm{P}\left(L^i \mid R^i \otimes \delta \otimes \pi\right) = \mathrm{f}_{\mu(R^i,\delta)}\left(L^i\right)$

- Identification *(of some of the terms of the decomposition)*

  => *Experimental Data* δ $\mathrm{P}\left(L^i \mid R^i \otimes \delta \otimes \pi\right) = \mathrm{P}\left(L^i \mid R^i \otimes \delta' \otimes \pi'\right)$

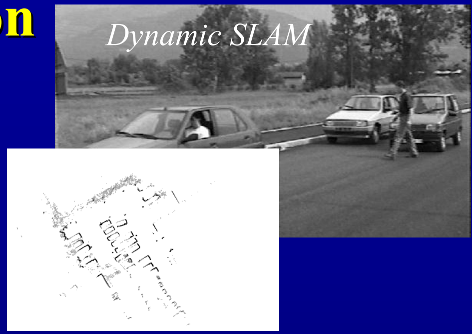
Question

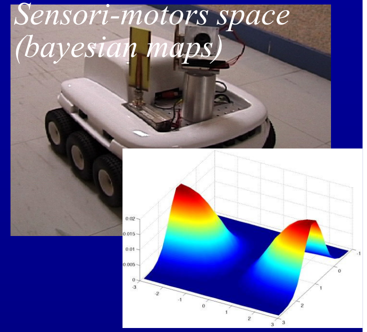
$$\mathrm{P}\left(Search \mid Known \otimes \delta \otimes \pi\right) = \frac{1}{\Omega} \times \sum_{Unknown} \mathrm{P}\left(X^1 \otimes X^2 \otimes \ldots \otimes X^n \mid \delta \otimes \pi\right)$$

=>Inference (i.e. answer to the question) carried out by an "inference engine" (symbolic simplification + numerical computation)

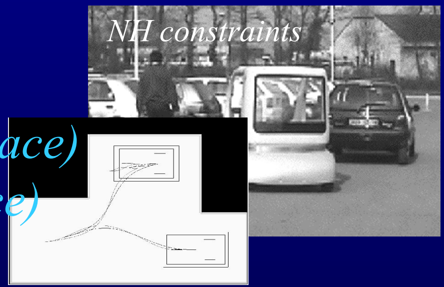
Research axes

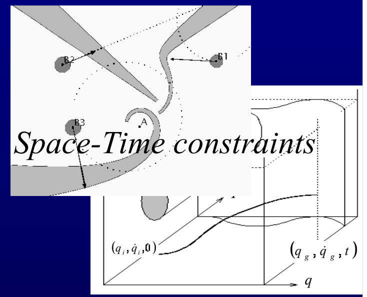
- **Multi-modal modeling of space & motion**
  - *Incremental world modeling*
  - *Prediction & estimation of obstacles motions*
  - *Sensori-motors maps*



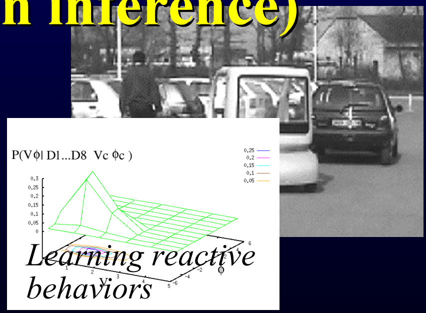


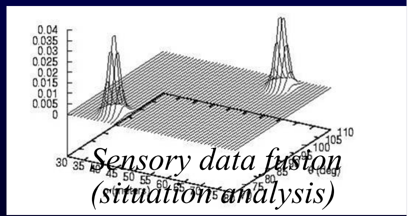
- **Motion planning in a dynamic world**
  - *Iterative trajectory planning (ITP)*
  - *Instantaneous escaping trajectories (Velocity space)*
  - *States of unavoidable collisions (State-time space)*





- **Decision in an uncertain world (Bayesian inference)**
  - *Bayesian programming*
  - *Automatic learning & Entropy maximization*
  - *Biological inspiration*

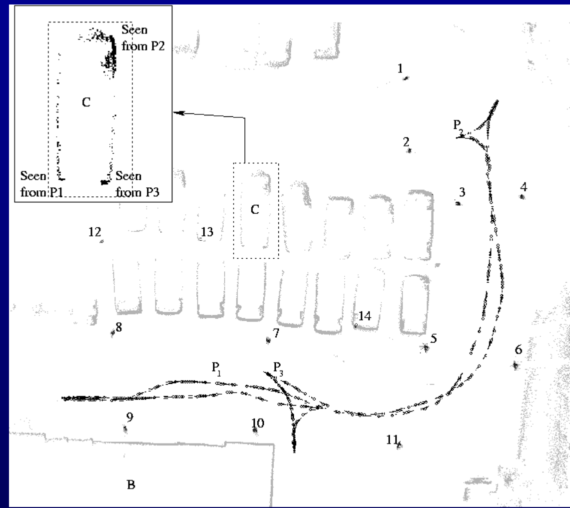




Autonomous navigation & Easy robot programming

⇒ Several functionalities (learned and downloaded) have to be combined

Incremental world modeling & localization + Motion planning + Autonomous sensor-based navigation



SLAM

+

Motion planning

+

Reactive navigation

[Pradalier & Hermosillo 03]

Bayesian programming of reactive behaviors

[Pradalier et al. 03]

=> Controlling the vehicle using a probability distribution on (v,ϕ)

e.g. reducing speed and/or modifying steering angle for avoiding a pedestrian or a car

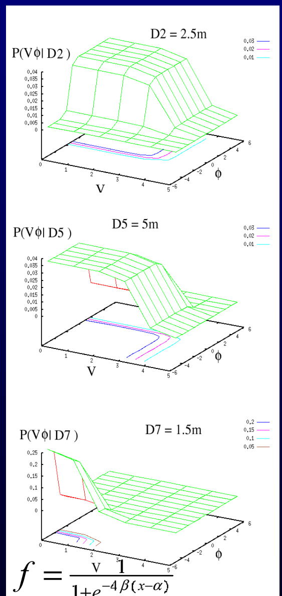
Joint distribution for the fusion :

$$P(V \otimes \phi \otimes D_1 \otimes \ldots \otimes D_8) = P(V \otimes \phi)\prod_{i=1}^{8} P_i(D_i / V \otimes \phi)$$

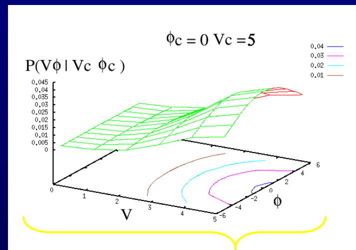
where :

$$\begin{cases} P(V \otimes \phi) = \text{Uniform} \\ P_i(D_i / V \otimes \phi) = \dfrac{P_i(D_i)P_i(V / D_i)P_i(\phi / D_i)}{\sum_{D_i} P_i(D_i)P_i(V / D_i)P_i(\phi / D_i)} \end{cases}$$

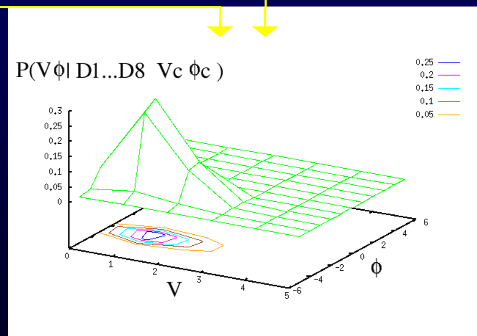
Probabilistic joint distribution for area i

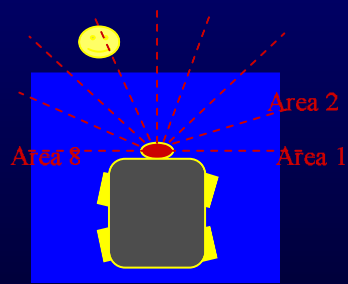


Command fusion



Gaussian function





Current work: *More complex situations & Learning, Integration with Planning & Control*

Bayesian programming : some experimental result

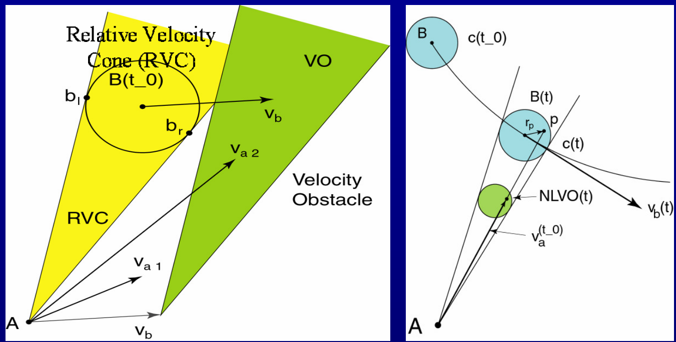
Reactive obstacle avoidance (Cycab)

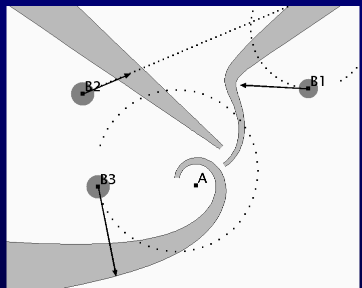
Target following (Koala)

V-Obstacles & ITP (dynamic environment)

[Large et al. 03]

- *Instantaneous escaping trajectories (V-obstacles) => Strategies for avoiding moving obstacles*
- *Iterative Trajectory Planning => Complete navigation strategy*



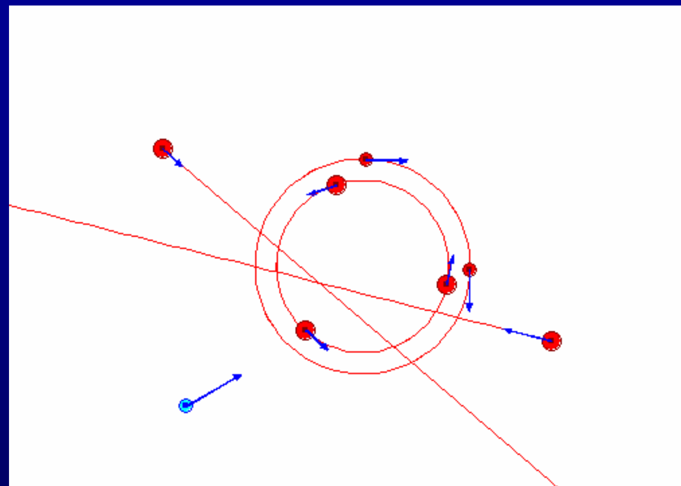


Obstacle trajectory : $c(t) = d(t) e^{i\theta(t)}$

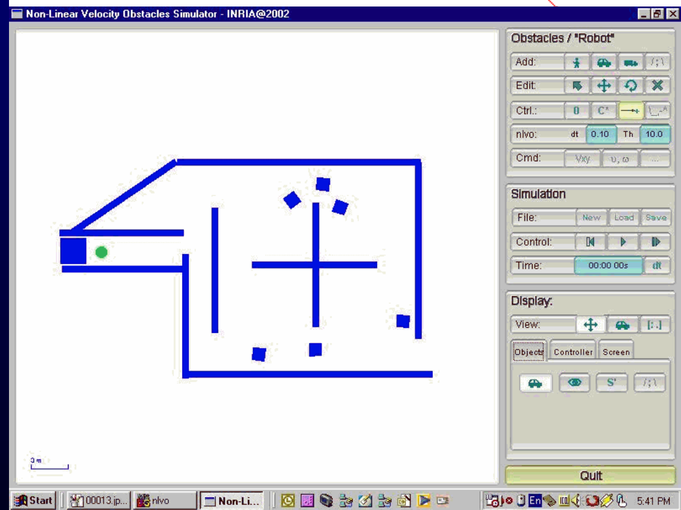
$c_v(t) = \frac{d(t)}{t} e^{i\theta(t)}$

$vo_r(t) = c_v(t) + i\frac{r}{t}\hat{c}_l(t)$

$vo_l(t) = c_v(t) - i\frac{r}{t}\hat{c}_l(t)$



V-obstacles

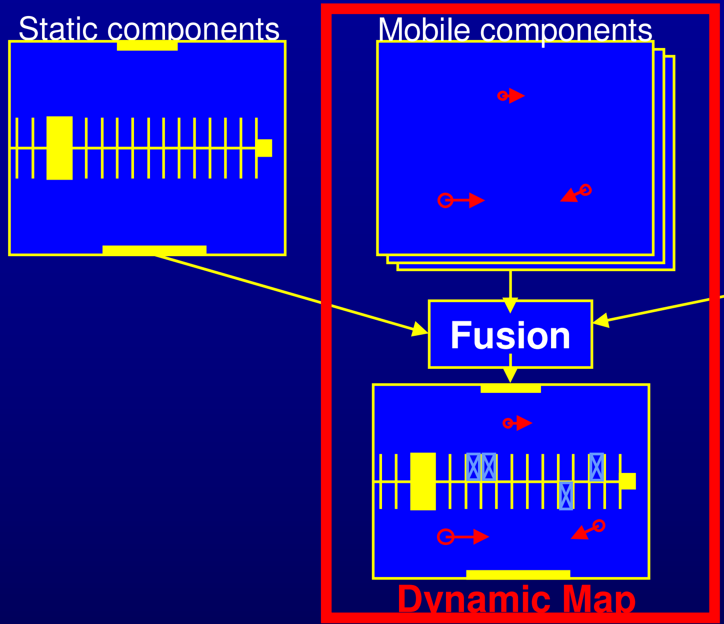


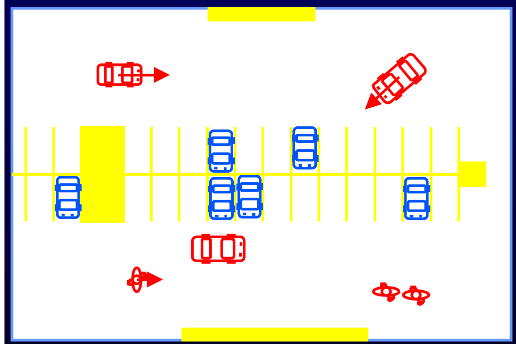
V-obstacles
+
ITP

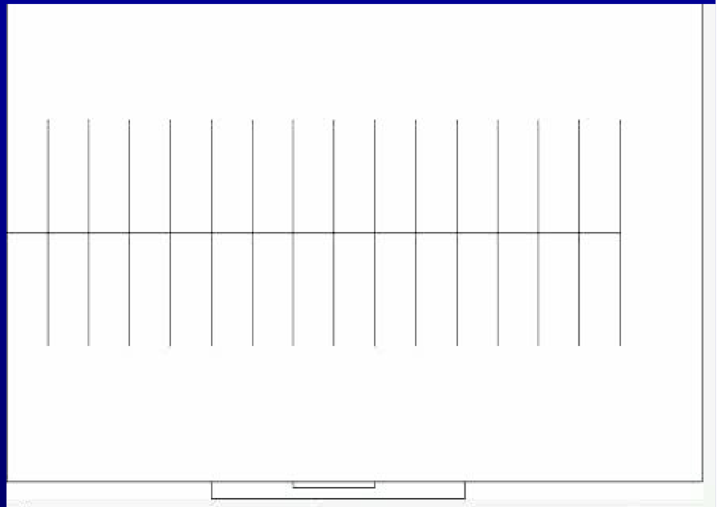
Current work: *More complex geometry & dynamics, Perception, Uncertainty*

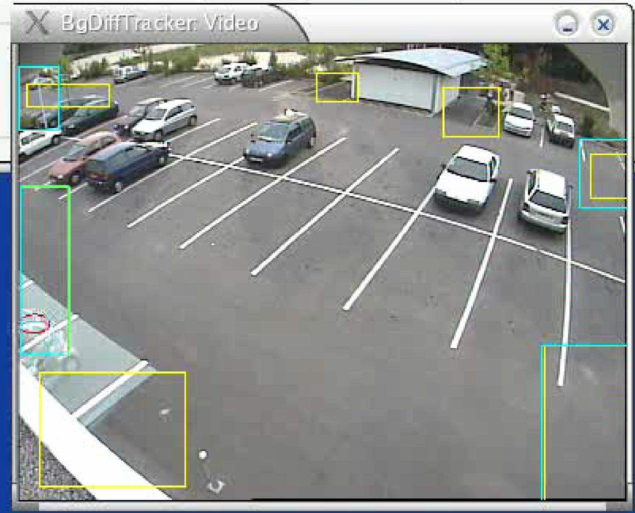
Automatic reconstruction of a dynamic map
(Parkview : Multi-camera system)

[Helin 03]



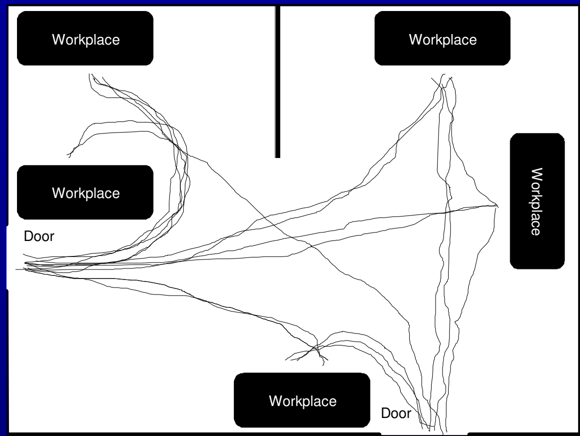






Trajectory prediction for moving obstacles

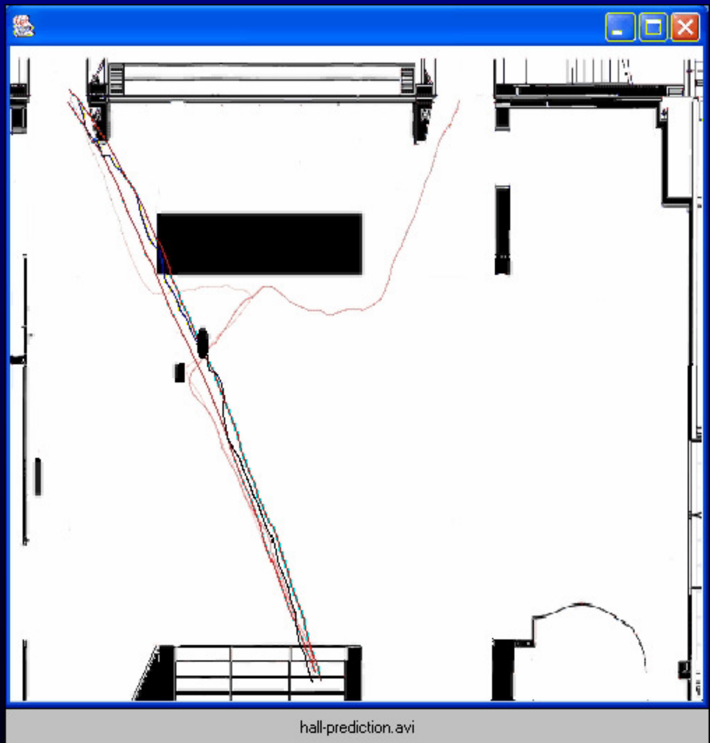
[Vasquez 04]



Learning phase : *Clustering*

Mean trajectory : $\mu_k(t) = \frac{1}{N_k}\sum_{i=1}^{N_k} d_i(t)$

Standard deviation : $\sigma_k = \left(\frac{1}{N_k}\sum_{i=1}^{N_k}\delta(d_i,\mu_k)^2\right)^{1/2}$



Prediction phase (at each time step) :
Calculating the likelihood that $d_p \in C_k$

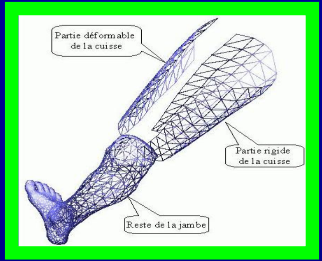
Partial distance : $\delta_p(d_p,d_j) = \left(\frac{1}{T_p}\int_{t=0}^{T_p}\left(d_p(t)-d_i(t)\right)^2 dt\right)^{1/2}$

Likelihood estimation : $P(d_p \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k}}\, e^{-\frac{1}{2\sigma^2}\delta_p(d_p,\mu_k)^2}$

Some previous results of our research team

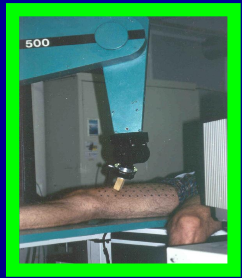
- *Interactive medical simulation*
- *Autonomous navigation for Virtual Reality applications*
- *Automatic driving & Driving assistance*

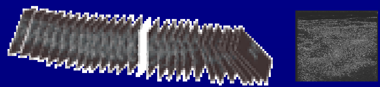
Interactive medical simulation



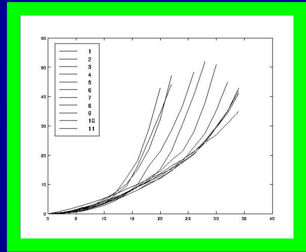
Geometric model

+





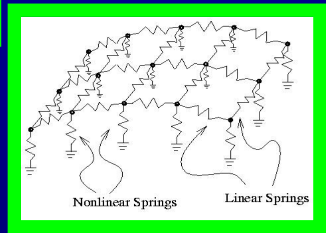
Measured data

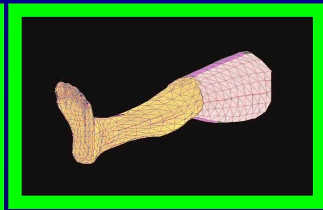


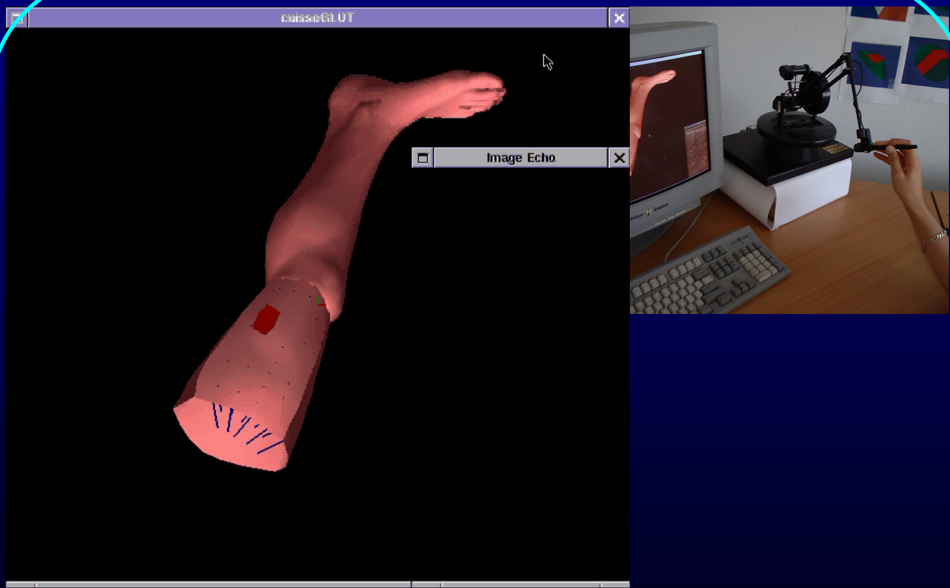
Stress-strain curves (measured)

$$F = k\Delta x \quad (linear)$$

$$F = \frac{\Delta x}{a\Delta x + b} \quad (non-linear)$$





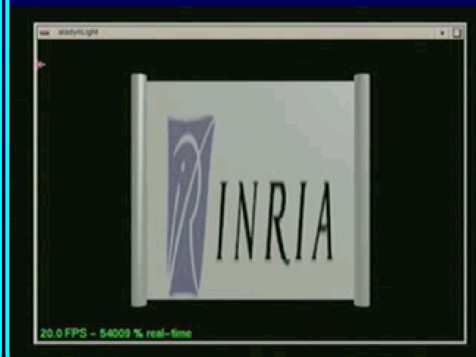


Echographic simulator *(coop. TimC & UC-Berkeley & LIRMM)*

[Daulignac & Laugier 00-01]

Separating the Gall-bladder from the liver

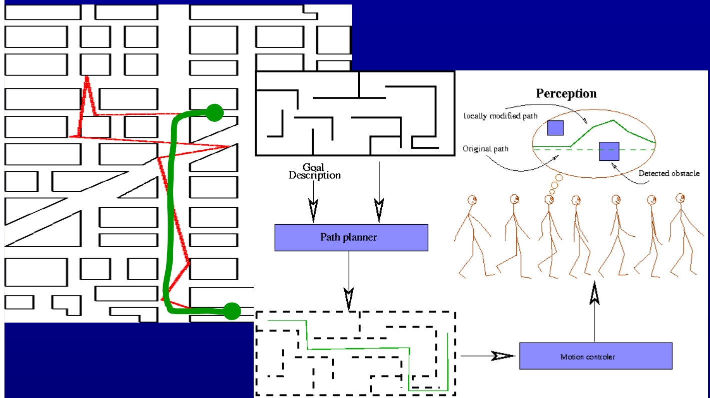
[Boux & Laugier & Mendoza 01]



Cutting a flag using an haptic device

[Boux & Laugier 00]

Autonomous navigation for Virtual Reality applications



- Dynamic path planning : *Adriane's Clew Algorithm [Ahuactzin 94]*
- Reactive navigation :
  => Path tracking & Obstacle avoidance [Raulo &Laugier 00]
  => Bayesian behaviors [Lebeltel 99, Raulo 01]

Question : P(M | S Cp_Surveil)

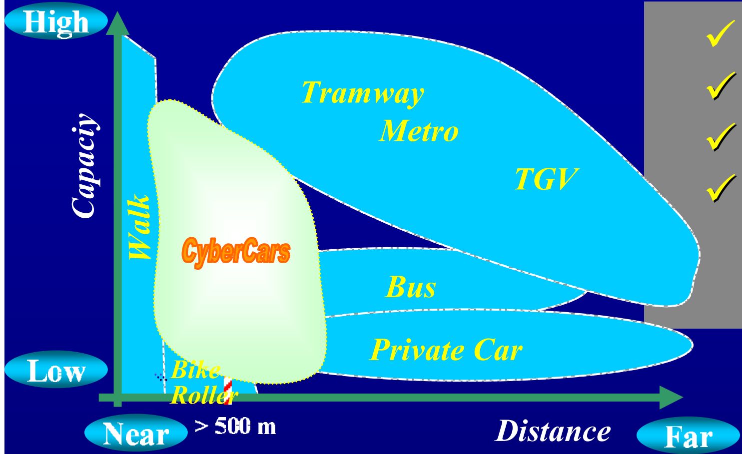
$$P\left(\mathtt{Vrot\ \ Vtrans} \;\middle|\; \begin{matrix}\mathtt{px0\ ..px7\ \ lm0\ ..lm7\ \ veille\ \ feu\ \ obj?} \\ \mathtt{eng\ \ tach\_t\ \ -1\ td\_t\ -1\ tempo\ \ tour} \\ \mathtt{dir\ \ prox\ \ dirG\ \ proxG\ \ \ vtrans\_c} \\ \mathtt{dnv\ \ mnv\ \ mld\ \ per}\end{matrix}\ \ Cp\_Surveil\right).$$

Solving : P(`Vrot Vtrans` *|* `px0 px1 … lm7 veille feu … per`*)*

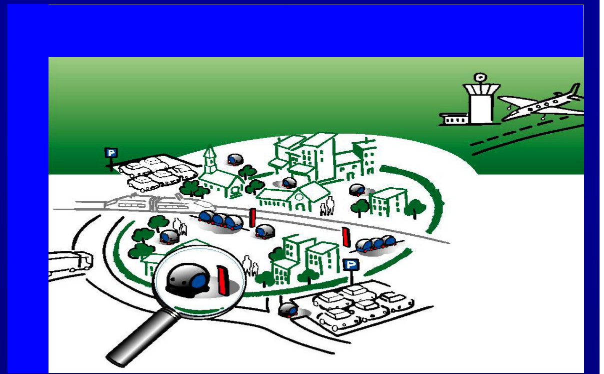
Some previous results of our research team

- *Interactive medical simulation*
- *Autonomous navigation for Virtual Reality applications*
- *Automatic driving & Driving assistance*

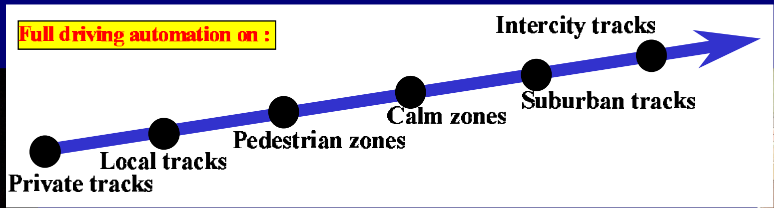
The « CyberCars » approach

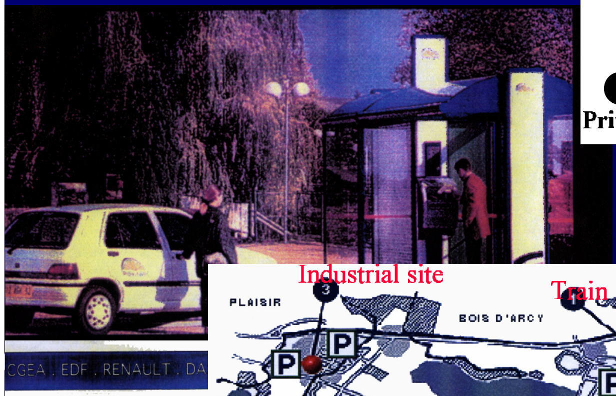


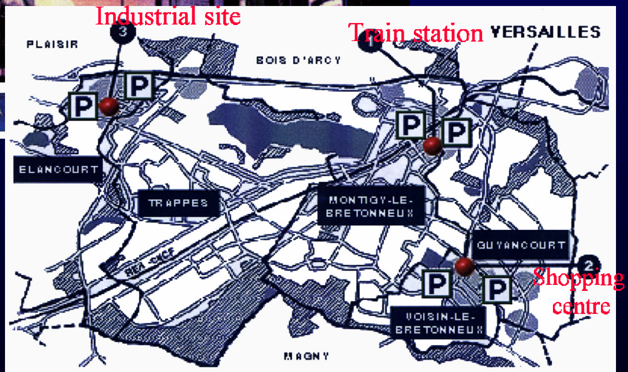
- ✓ Door to door, 24 hours a day
- ✓ Small (urban size), silent
- ✓ User friendly interface
- ✓ Automatic manoeuvres
  => parking, platooning
  ... up to fully automated



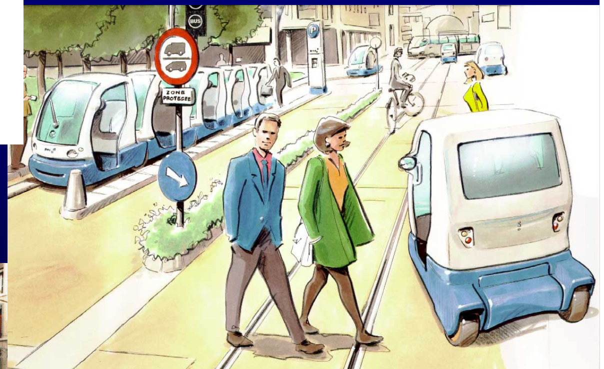
CyberCars are focusing on historical city centres







Praxitele : Real experiment in SQY (97-99)



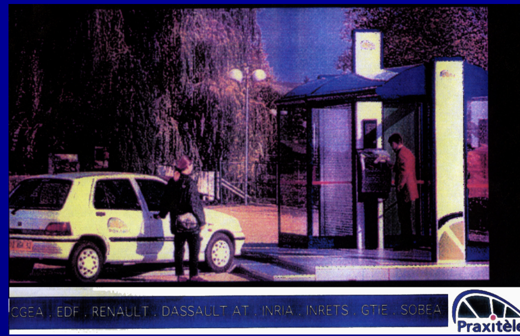
CyCab dual-mode vehicle

Commercialized by Robosoft

Some CyberCars projects

ParkShuttle (Frog, Netherlands)

Serpentine (Switzerland)



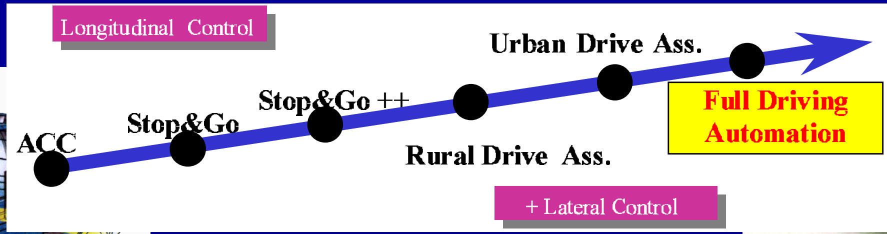
Praxitele (France)

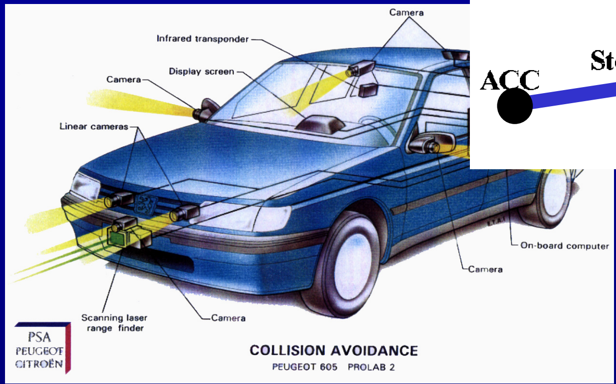
European Cybercars project (2001-05)

- ✓ **10 industrial partners** *(Fiat, Yamaha, Frog …)*, **7 research institutes** *(Inria, Inrets, Ensmp …)*, **12 cities involved** *(Rome, Rotterdam, Lausanne, Antibes …)*
- ✓ **10 M €**

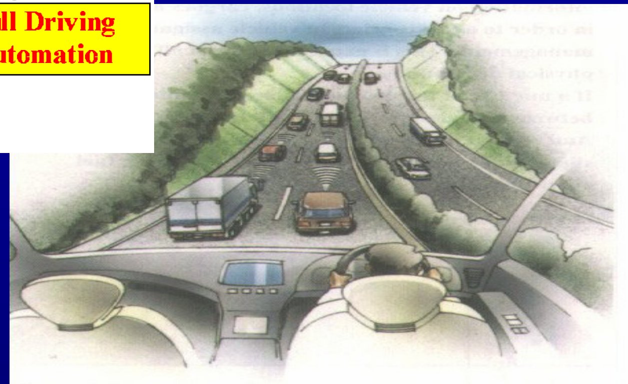
The « Automotive » approach

ADAS : Advanced Driver Assistance



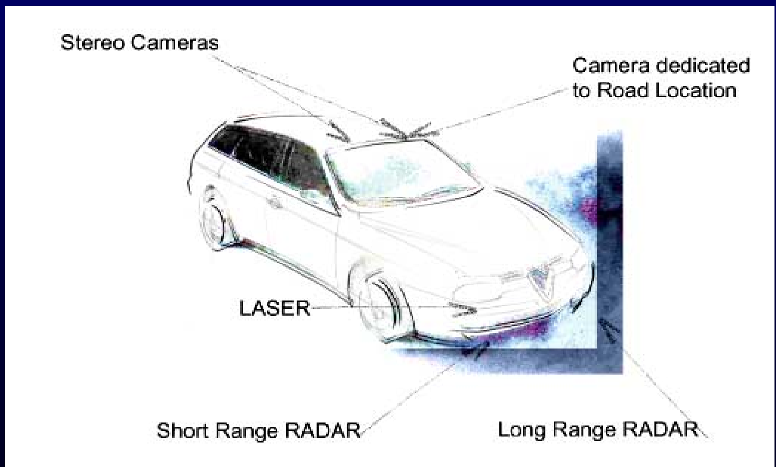


European Prometheus project (86 - 94)

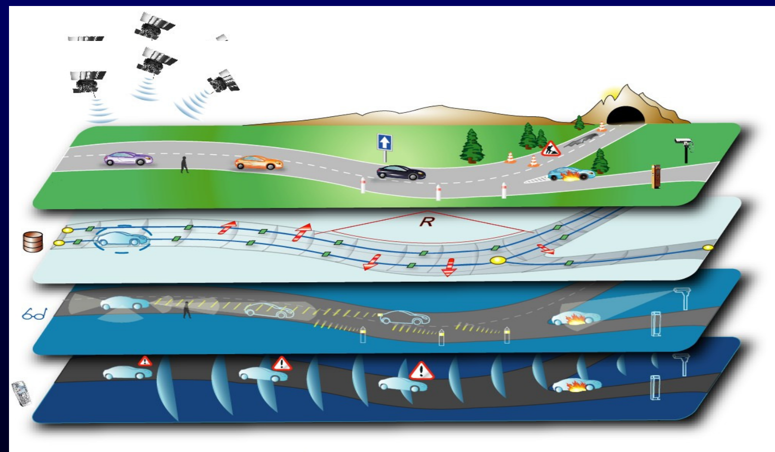


Current projects: Carsense, Arcos, Prevent

R&D program (on-board and off-board systems) for increasing safety & driving confort



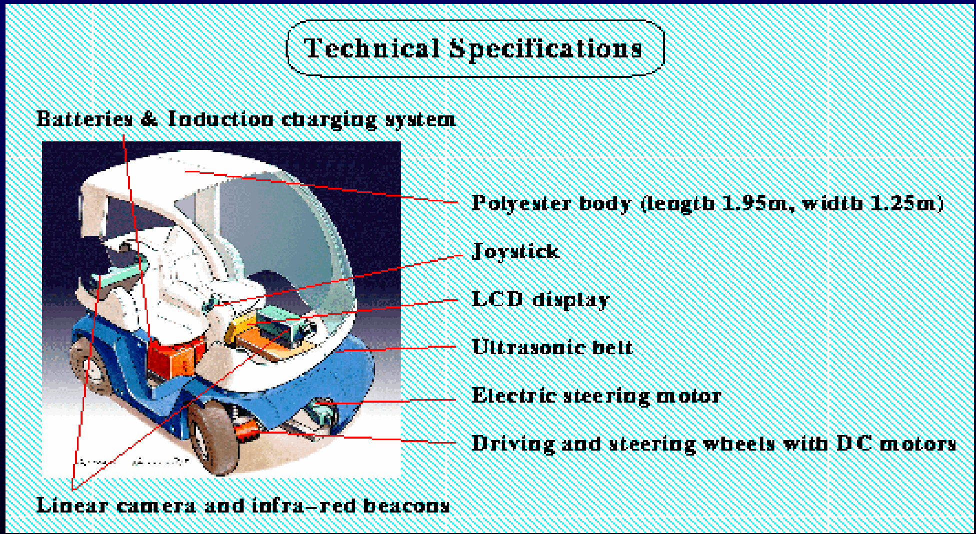
Carsense (car manufacturers & suppliers)
Sensor fusion for danger estimation



French Arcos project:
Vehicle - Infrastructure - Driver systems for road safety

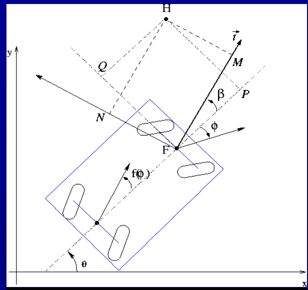
Experimental vehicles at INRIA Rhône-Alpes

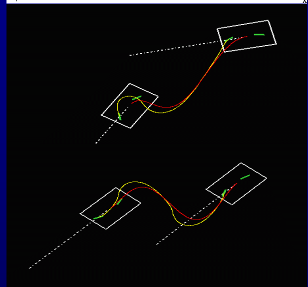
- electric vehicle with front driven and steering wheels
- abilities of human or computer-driven motion
- control system: VME CPU-board, transputer net
- sensor : odometry, ultrasonic sensors, linear CCD camera



Models for controlling the Cycab

$$\begin{pmatrix}\dot{x}_R\\ \dot{y}_R\\ \dot{\theta}\\ \dot{\phi}\end{pmatrix} = \begin{pmatrix}\cos(\theta+f(\phi))\\ \sin(\theta+f(\phi))\\ \dfrac{\sin(\phi-f(\phi))}{L\cos(\phi)}\\ 0\end{pmatrix} u_1 + \begin{pmatrix}0\\ 0\\ 0\\ 1\end{pmatrix} u_2$$





Differential Flatness of the Cycab

Turning frame : $(F, \vec{t}, \vec{t}^{\perp})$ *with* $\beta(\phi) = \tan^{-1}\frac{B(\phi)}{A(\phi)}$

$$\vec{t} = \cos\phi\, f'(\phi)\,\vec{u}_{\theta+\phi} - \cos f(\phi)\,\vec{u}_{\theta+f(\phi)}$$

$$A(\phi) = \cos^2(\phi) f'(\phi) - \cos^2 f(\phi)$$

$$B(\phi) = \cos(\phi)\sin(\phi) f'(\phi) - \cos(f(\phi))\sin(f(\phi))$$

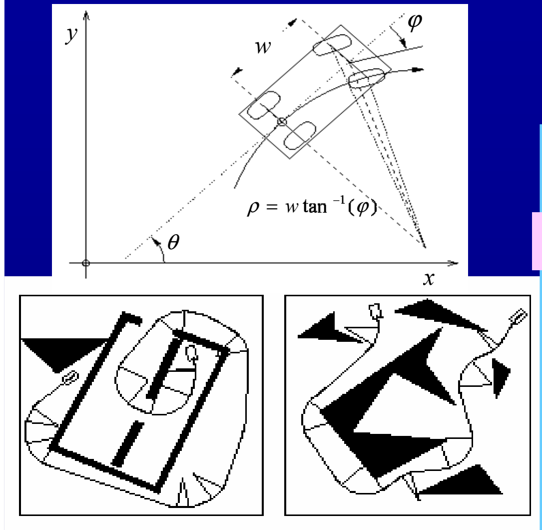
$$M(\phi) = \frac{L\cos^2(f(\phi))}{\sqrt{A^2(\phi)+B^2(\phi)}}$$

$$N(\phi) = -\int_0^{\phi} \frac{L\cos^2(f(u))(B'(u)A(u) - A'(u)B(u}{\left(A^2(u)+B^2(u)\right)^{3/2}}$$

- Existence (Differential flatness) *[Sekhavat, Hermosillo, 99]*
- Necessary conditions (« flat outputs ») *[Sekhavat, Rouchon, Hermosillo 01]*
- Analytic determination of the « flat outputs » *[Sekhavat, Hermosillo, Rouchon 01]*
- Current work :
  - Application to motion planning & autonomous navigation *(Hermosillo 03, Pradalier ...)*
  - Dealing with dynamic environments *(Pradalier, Fraichard, Petti, Vasquez ...)*

INRIA RHÔNE-ALPES

Decisional & Control architecture

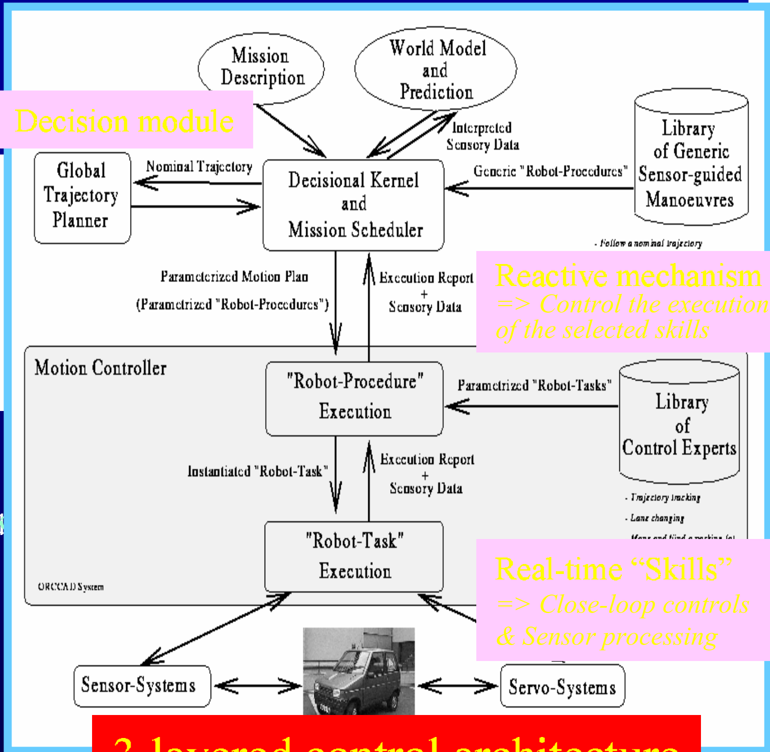


Planning CC-paths

(kinematic constraints ...)

continuous curvature profile + upper-bound curvature & curvature derivative

[Scheuer & Laugier 98]

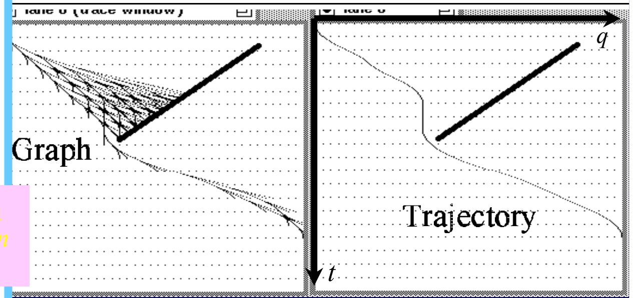


3-layered control architecture

[Laugier et al. 98]

$$\begin{cases} F_{min} \le F \le F_{max} & \text{(engine force limit)} \\ \sqrt{R_t^2+R_n^2} \le \mu R_b & \text{(no sliding constraint)} \\ 0 \le \dot{s} \le \dot{s}_{max} & \text{(speed limit)} \end{cases}$$

$$\begin{cases} 0 \le \dot{s} \le \min(\dot{s}_{max}, \sqrt{\frac{\mu g}{|\kappa_s|}}) \\ \max(\frac{F_{min}}{m}, \sqrt{\mu^2 g^2 - \kappa_s^2 \dot{s}^4}) \le \ddot{s} \le \min(\frac{F_{max}}{m}, \sqrt{\mu^2 g^2 - \kappa_s^2 \dot{s}^4}) \end{cases}$$



Kinodynamic Motion Planning

(Dynamic constraints ...)

[Fraichard 92]

Platooning [Parent & Daviet 96]

Lane Changing & Obstacle avoidance

[Laugier et al. 98]

Automatic Parallel Parking

[Paromtchik & Laugier 96]

INRIA RHÔNE-ALPES

« Platooning » [Parent & Daviet 96]

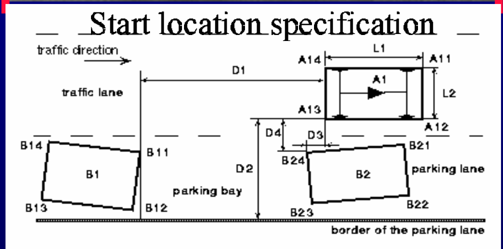
Electronic « Tow-bar »

CCD Linear camera + Infrared target
(high rate & resolution)

Automatic parking maneuvers

[Paromtchik & Laugier 96]

On-line local world reconstruction
& Incremental motion planning



$$\begin{cases} \phi(t) = \phi_{\max}\, k_\phi\, A(t), & 0 \le t \le T \\ v(t) = v_{\max}\, k_v\, B(t), & 0 \le t \le T' \end{cases}$$

$\phi_{\max} > 0,\ v_{\max} > 0,\ k_\phi =$

$$A(t) = \begin{cases} 1, & 0 \le t < t' \\ \cos\dfrac{\pi(t-t')}{T^*}, & t' \le t \le T - t', \\ -1, & T - t' < t \le T \end{cases} \qquad t' = \frac{T - T^*}{2}, T^* < T$$

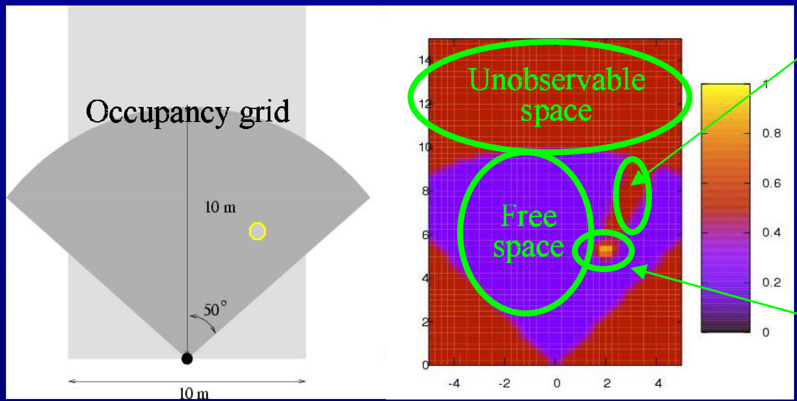
$$B(t) = 0.5\left(1 - \cos 4\pi t / T\right), \qquad 0 \le t \le T$$

=> On line motion planning using sinusoidal controls $\phi(t)$ and $v(t)$

(search for control parameters T and ϕmax)

INRIA RHÔNE-ALPES

Robust obstacles tracking
(Bayesian Occupancy Filter approach)

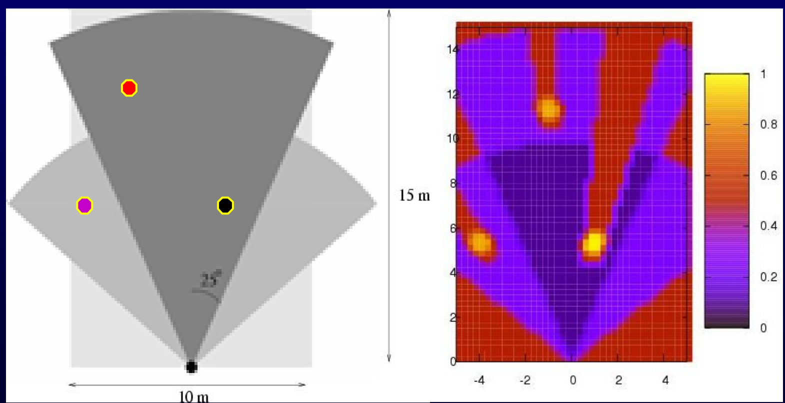


Concealed space

Occupied space

1 sensor & 1 object

P($[E_c=1]$ | z c)
$c = [x, y, 0, 0]$ and z=(5,2,0,0)



2 sensors & 3 objects

P($[E_c=1]$ | $z_{1,1}\, z_{1,2}\, z_{2,1}\, z_{2,2}\, c$)
$c = [x, y, 0, 0]$

$z_{1,1}$ = (5.5, -4, 0, 0) $z_{1,2}$ = (5.5, 1, 0, 0)
$z_{2,1}$ = (11, -1, 0, 0) $z_{2,2}$ = (5.4, 1.1, 0,0)

Program

Description

- **Specification**
  - Variables :
    - C : cell
    - E_C : cell occupancy (E_C=1 *means"occupied*
    - $Z_{1:S}$: observations
    - $M_{1:S}$: association (1 for each sensor)
  - Decomposition :

$$P(C\ E_C\ M_{1:S}\ Z_{1:S})$$
$$= P(E_C\ C)\prod_{s=1}^{S}\left(P(M_s)\prod_{i_s=1}^{O_s} P(Z_{s,i_s} \mid E_C\ C\ M_S)\right)$$

  - Parametric form :
    - P(E_C C) : *a priori* uniform
    - P($Z_{1:S}$ | E_C C) : sensor models
- **Identification** => *Calibration*

Question

- **Utilization** For all c : P(E_c | $Z_{1:S}$ c)

INRIA RHÔNE-ALPES

Robust obstacles tracking & avoidance
Experimental results with the Cycab

Prediction

$P(E_C^i \; C^i \mid z^{0:i-1} \; a^{0:i-1})$

z^i

Estimation

$P(E_C^i \; C^i \mid z^{0:i} \; a^{0:i-1})$

Bayesian Occupancy Filter (BOF)

Pedestrian avoidance using the BOF